

RESEARCH ARTICLE

8T-SRAM Based Process-In-Memory (PIM) System With Current Mirror for Accurate MAC Operation

KEUN-YONG CHUNG, HONGGU KIM^{ID}, (Graduate Student Member, IEEE),
YERIM AN, (Graduate Student Member, IEEE),
KIHO SEONG^{ID}, (Graduate Student Member, IEEE),
DONG-HYUN SHIN^{ID}, (Graduate Student Member, IEEE),
KWANG-HYUN BAEK^{ID}, (Senior Member, IEEE), AND YONG SHIM^{ID}, (Member, IEEE)

School of Electrical and Electronic Engineering, Chung-Ang University, Dongjak-gu, Seoul 06974, Republic of Korea

Corresponding authors: Kwang-Hyun Baek (kbaek@cau.ac.kr) and Yong Shim (yongshim@cau.ac.kr)

This work was supported in part by the Chung-Ang University Graduate Research Scholarship, in 2022; and in part by the National Research Foundation of Korea (NRF) Grant funded by the Korean Government through Ministry of Science and ICT (MSIT) under Grant 2021R1C1C1008752 and Grant 2021M3F3A2A0103773.

ABSTRACT Process-in-memory (PIM) is an emerging computing paradigm to overcome the energy bottleneck inherent in conventional computing platform. While PIM utilizes several types of memory elements, SRAM based PIM has been researched extensively for its high scalability and feasibility by using CMOS process technology. In this work, we proposed 8T SRAM based process-in-memory system with current mirror for accurate MAC operation. To resolve the nonlinearity issue inherent in current-based SRAM MAC macro, we utilized cascaded current mirror. Furthermore, to enable more precise MAC operation, we allocated RWL timing pulses which drive 8T SRAM bitcells in the middle of timing duration. Our PIM architecture realize 4 bit input, 4 bit weight and 4 bit output precisions. Prototype chip has been fabricated using TSMC 65nm GP process technology. Measurement result has proved the MAC operation linearity of designed 8T SRAM based PIM system, and its throughput and energy efficiency with 0.29 GOPs and 1.05 TOPs/W. Software level analysis proved that our design can achieve upto 98.02 percent accuracy for MNIST dataset classification and 94.93 percent accuracy for CIFAR-10 dataset classification.

INDEX TERMS Processing-in-memory (PIM), current-mode accumulation, current mirror, multiply-and-accumulate (MAC), multi-bit computing, static random access memory (SRAM).

I. INTRODUCTION

In the recent era, Deep Neural Network (DNN) has kept on development for its huge effectiveness when it is applied to broad range of applications such as visual recognition, fraud detection, healthcare and self-driving cars. As the DNN task has become more complex in the real-life application, the DNN model has become deeper and bigger, to obtain better performance in classification accuracy. However in return, the number of data to be processed and moved has exponentially increased, incurring von Neumann bottleneck [1].

Von Neumann bottleneck is mainly due to tremendous data movement between processing units (e.g. CPU) and memory

units (e.g. SRAM, DRAM) [2], [3]. Since, these two components are separated, the parameters and input data stored in memory units should be moved and fetched to processing units for DNN inference. This tremendous data movement leads to tremendous energy consumption. Furthermore, when considering simple and repeated form of DNN inference computation, taking this energy penalty into account to deal with very simple DNN tasks is not appropriate [3], [4].

To make a breakthrough in this energy problem, new paradigm of computing platform has emerged called Process-in-Memory (PIM). The main feature of PIM is collocated processing and memory units, which eliminates tremendous data movement between these units and their power consumption. Moreover, this scheme is specifically appropriate when it is applied to simple and repeated DNN operation

The associate editor coordinating the review of this manuscript and approving it for publication was Chao Wang^{ID}.

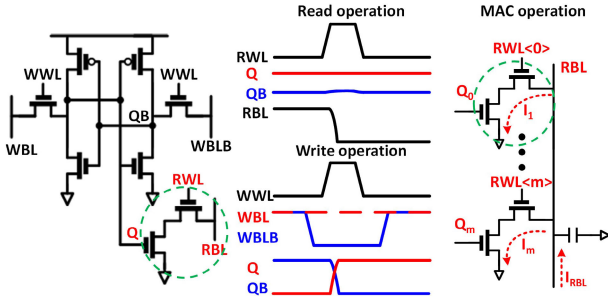


FIGURE 1. Conventional 8T-SRAM architecture and its operation mode.

(e.g. Multiply-and-Accumulate). Consequently, many researches have proved the efficacy of PIM architecture when it replaces von Neumann architecture [5], [6], [7], [8], [9], [10], [11], [12], [13].

SRAM cell is the most popular device for implementing PIM architecture due to its simple operating principle and the feasibility of using by the typical CMOS process technology [14], [15], [16], [17], [18], [19], [20], [21]. In this work, we use the conventional 8T-SRAM, where the bit-cell structure and read/write operation, along with the basics of MAC operation are shown in Fig. 1. During the write operation, the voltage levels on word-line and word-line bar (WBL/WBLB) are transferred to the storage node Q/QB when the Write Word Line (WWL) is high. During the read operation, the value stored on Q determines the current flow through a series of transistors in the read path when the Read Word Line (RWL) is driven to logic high.

Meanwhile, the MAC operation includes the summation of the multiplication outputs, where the multiplication typically occurs within the bit cell, while the summation is performed by the current flow from/to the Read Bit Line (RBL) using the Kirchhoff Current Law (KCL). Considering the binary product between the data stored in bit cell (Q_m) and the logic level of the RWL switch $RWL<m>$, the read current (I_m) discharges the precharged RBL only when the multiplication output becomes 1, otherwise the RBL maintains its precharged value. The amount of current flow (I_{RBL}) during the MAC operation is linearly proportional to the number of multiplication output '1's among the bit cells connected to the same RBL.

However, the above MAC operation only performs binary multiplication, not the multi-bit multiplication typically required for most real-world application. To solve this issue, several techniques are actively researched [5], [6], [7], [9], [10], [14], [15], [16], [17], [18], [19], [20], [21], [22], [23], where some of the main ideas with 4-bit resolution are illustrated in Fig. 2. Most common approaches to multi-bit expressions based on SRAM bit cells can potentially group multiple bit cells and assign MSB to LSB accordingly. The case I shown in Fig. 2 is one such example, where the read access transistors are sized in a binary weighted manner so that the read current changes linearly depending on their sizes. Other common approaches include 1) pulse-width modulation (PWM) in case II which controls the read

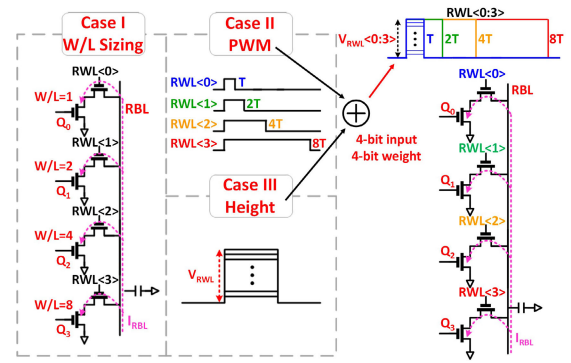


FIGURE 2. Multi-bit expression techniques and 4-bit MAC operation.

enable time in a binary weighted manner and 2) pulse-height modulation (PHM), which controls the gate voltage of the read access transistor to control the conductivity (case III). Some of the techniques can also be combined into a single system where the PWM (case II) and PHM (case III) are mixed together to modulate the shape and duration of the RWL signal. Then, two factors to be multiplied are mapped onto a single signal, and the size of the read access transistor of the 8-T SRAM could be maintained, which would compromise the compact layout of the SRAM bit cell.

When considering the circuit-level implementation of the PWM and PHM methods, the former requires simple combinational logic such as a shift register and some logic gates. However, the latter PHM method typically uses a digital-to-analog converter (DAC), which requires a resistor ladder and a unit-gain buffer. In addition, maintaining a fine voltage step and providing a stable voltage level becomes more difficult as the number of bits to express the data increases. The capacitors at each branch of the R-ladder and the power-hungry operational amplifier inside the unit-gain buffer add to the area and power consumption.

To reconcile such design issues, and also to provide fine controllability of the multiplication operation, this article proposed a current mirror-based MAC operation. By adjusting the size ratio of the source and binary weighted current branches, the exact amount of the current flowing at each SRAM bit cell during the read operation could be obtained. In addition, the amount of current at the read access transistor is adjusted only by controlling the source current mirror, which relaxes the area overhead of the read access transistor with binary weighted sizes in the case I.

II. PROPOSED SRAM MAC HARDWARE ACCELERATOR

The core schematic of the proposed current mirror-based MAC computation unit is shown in Fig. 3. The unit is basically divided into two parts: "Input Current Mirror" and "Additional Current Branch".

A. MULTI-BIT MULTIPLICATION OPERATION VIA CURRENT MIRROR

The input current mirror consists of 1) bias voltage generator (Ref. Current), 2) 4-bit binary weighted current branch,

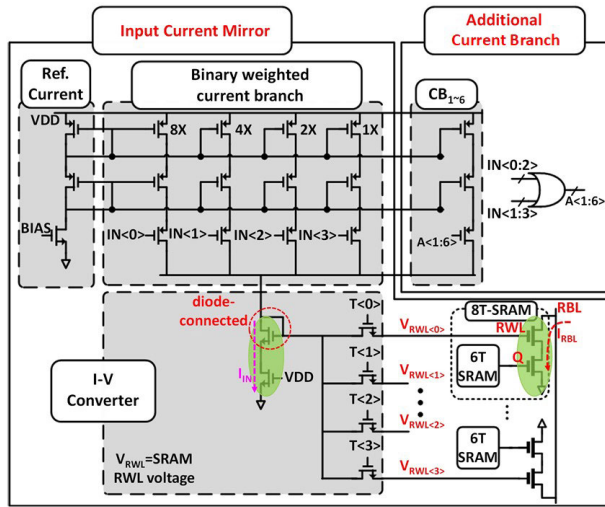


FIGURE 3. Proposed 4-bit input current mirror based PIM architecture.

3) I-V converter, and 4) 8T SRAM cell. In this subsection, 1) the MAC operation based on the proposed current mirror-based PIM architecture, and also 2) the physical quantities that map two multiplication factors are explained in detail.

The bias voltage generated by the beta multiplier reference unit is used to set the unit current of the Binary weighted current branch by driving the upper two series transistors of the current mirror. There are also switch transistors controlled by $IN<3:0>$ at the bottom of the current mirror that controls the final current flow (I_{IN}) through the bottom diode-connected transistor. Thus, the amount of current controlled by 4-bit input signal represents the first physical factor used to map a multiplicand of a multiplication operation.

The current I_{IN} is then converted into a bias voltage by the I-V converter, which eventually controls the conductivity of the read access transistor in each bit cell when the intermediate switching transistors with $T<3:0>$ signals are turned on. Here, the $T<3:0>$ signals are generated by the PWM unit, where the MSB signal $T<3>$ has a pulse width 8 times longer than the LSB signal $T<0>$. These 4 control signals and the corresponding 4-bit cells form the second physical factor to represent the multiplier (or weight) of a multiplication.

Note that the diode-connected transistor in the I-V converter and the read access transistor in each bit cell form another sub-current mirror circuit, so the read current flow basically follows the original current flow from the current mirror (I_{IN}) when two transistors have the same size. There is also a bottom NMOS transistor in the IV converter that mimics the bottom transistor in the read path inside the SRAM core. This additional transistor balances the source branch and the destination branch of the sub-current mirror for better linearity.

However, the different V_{DS} voltages on the source branch and the binary weighted current branches causes the different current flow. To address this nonlinearity problem, an

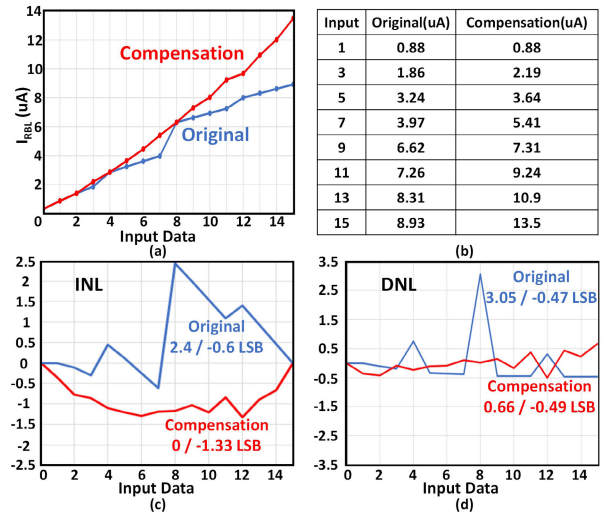


FIGURE 4. (a) Measured I_{RBL} plot, (b) measured I_{RBL} table, static performance of (c) INL and (d) DNL with/without additional current branch.

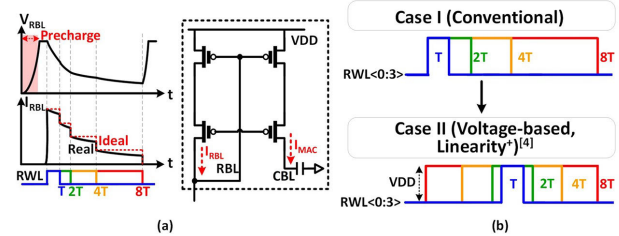


FIGURE 5. (a) Cascade Current Mirror (CCM) circuit for discharging issue, and (b) Prior PWM.

additional current branch is added to the binary weighted current mirror stage. The additional control code $A<1:6>$, which is generated from a simple logic with $IN<3:0>$ signal, calibrates the nonlinear current flow by controlling the current flow from the additional current branch circuit. The effect of the calibration unit on the I_{RBL} current, INL, and DNL is shown in Fig 4. Compared to the original current flow based on the 4-bit input signal ($IN<3:0>$), the current flow with the compensation unit shows much better linearity.

Finally, it is worth noting that two multiplication factors are mapped to the amount of current (dictated by the current mirror) and to the read enable time duration of 4 bit cells in a single column. Therefore, the bit cell transistors have the minimum width and length for better area efficiency.

B. CASCADE CURRENT MIRROR AND PWM CIRCUIT

The multiplication output generated in each bit cell is eventually represented by the read current, where the charge moves from the precharged read bit line (RBL) to ground through the series of read transistors. This discharging process takes place for the predetermined period of time dictated by the RWL. Even though we expect the discharge current to remain stable during the read operation, the voltage drop from the RBL as the charge moves causes the non-linear current flow as shown in Fig. 5(a). To address this

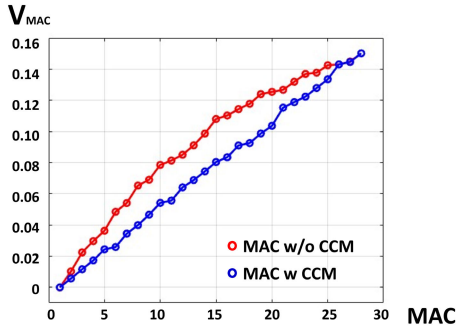


FIGURE 6. Enhanced linearity with cascade current mirror technique.

issue, instead of directly discharging the read current from the RBL capacitors (including the parasitic and additional RBL capacitors), the cascade current mirror (CCM) circuit from [21] is adopted. Note that the CCM circuit here basically decouples the original RBL from the charge bit line (CBL), and helps current become stable during the read operation due to the unwanted voltage fluctuation. Moreover, the ratio between I_{RBL} and the copied current to the CBL (I_{MAC}) can be easily adjusted by changing the size of the left/right current mirror transistors. This makes the I_{MAC} current lower than its original branch (I_{RBL}), resulting in a smaller RBL capacitor for a smaller area, and also supports the linearity of the MAC operation even with a longer read time.

Fig. 6 shows one column 8T SRAM MAC operation with and without cascade current mirror implementation. Note that pulse width modulation has not been adopted in this analysis, and the pulse width has been fixed for all 8T SRAM RWL lines, and MAC voltages on CBL were analyzed from 0 to 30 MAC values. The non-linearity analysis can be explained more explicitly with the aid of following equation.

$$I_d = \frac{1}{2} \mu_n C_{ox} \frac{W}{L} \left((V_{gs} - V_{th}) V_{ds} - \frac{V_{ds}^2}{2} \right) \quad (1)$$

where drain source voltage can be expressed as follows.

$$V_{ds} = (V_{gs} - V_{th}) + \sqrt{(V_{gs} - V_{th})^2 - 4 \frac{I_d}{\mu_n C_{ox} \frac{W}{L}}} \quad (2)$$

In this case, V_{gs} is kept with constant value. Without cascade current mirror, the increment of V_{ds} decreases with increasing MAC values, causing the 8T SRAM bitcell current gradually decreases. On the contrary, after adopting cascade current mirror, consistent V_{ds} is guarantee and the resulting MAC voltages exhibits enhanced MAC operation linearity.

Apart from the CCM circuit, the initiation time of the binary weighted RWL pulses has been modified in the proposed design. The typical RWL pulse generation method assigns the read enable pulse with the minimum width corresponding to the LSB signal (i.e. RWL<0> in our design) early in the timing window, as shown in the Case I of Fig. 5(b). However, the role of the RWL pulse in our design allows the current to flow through the read access transistor

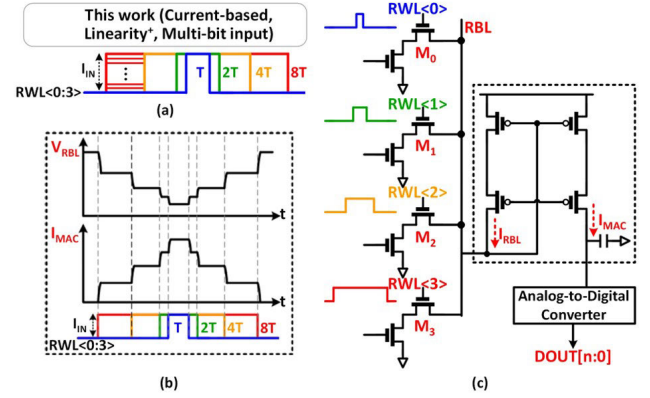


FIGURE 7. (a) Implemented PWM, (b) Constant I_{MAC} during MAC operation, and (c) Column-wise MAC operation architecture.

by transferring the V_{RWL} level to the read access transistor. Accordingly, it takes some time to stabilize the V_{RWL} level initially, and this will slightly delay the normal flow of the read current at the beginning of the operation, resulting in a deviation from the exact calculation results. To alleviate the similar problem, the previous work in [21] allocates the shorter enable pulse in the middle of the read timing window, as illustrated in Case II of Fig. 5(b), where the middle part typically provides the most stable operation.

By combining the above techniques, Fig. 7 shows the core part of the MAC computation unit proposed in this article. Note that the RWL<3:0> pulses are initiated in the different timing as in [21], but the height of the voltage represents the bias voltage level from the source current mirror stage (depends on the one multiplication factor), which controls the amount of current flow during the MAC operation. The simulation results depicted in Fig. 7(b) also show that the proper timing of the RWL pulses (when all the pulses are enabled sequentially) leads to a stable transition in terms of V_{RBL} and the corresponding I_{MAC} level. Finally, the column-wise connection of the 4bit multiplier along with its peripheral circuits, such as CCM, RBL capacitor, and ADC is shown in Fig. 7(c).

C. OVERALL ARCHITECTURE

The overall PIM architecture based on the proposed current mirror-based MAC computation unit is shown in Fig. 8. The memory array contains 16×8 8-T SRAM bit cells inside, where a single column has 16 bit cells and simultaneously performs column-wise four 4-bit multiplication operations. The multiplication result of each bit cell is then accumulated along an RBL to complete the MAC operation. The resulting I_{RBL} current passes through the CCM and charges the RBL capacitor, which is converted into the 4-bit digital code via the ADC. ADC has been designed with 4 bit FLASH ADC. Though FLASH ADC commonly exhibits large power consumption, but it enables highly enhanced throughput when it is applied to PIM architecture for its one-cycle-only operation. In order to match the layout width of the bit cell

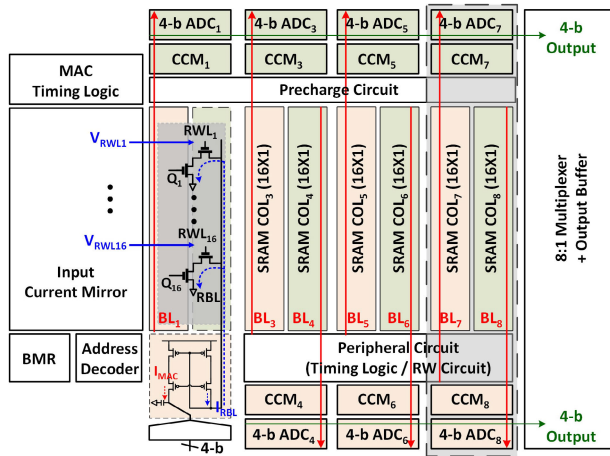


FIGURE 8. Overall architecture of the proposed PIM system.

array and the peripheral unit, the CCM unit and ADC have 2 column widths, and the peripheral unit is placed at the top and bottom of the column for efficient layout. Apart from the main column and peripheral units, the timing logic, reference current mirror, and address decoder unit are positioned on the left side of the chip, while the outputs from the ADCs are presented to the 8:1 multiplexer on the right side, so that an ADC output code could be checked externally to reduce the pin count.

Note that the input, weight and output bit precisions have been constrained to 4 bit in this work, to maximize the energy benefits. However, it may lead to accuracy drop when it is applied to machine learning classification task. However, when utilizing Quantization Aware Training (QAT), the accuracy drop is much alleviated as is discussed in section III.

III. EXPERIMENTAL WORKS

A. PROTOTYPE CHIP TEST RESULTS

The prototype MAC hardware accelerator is fabricated in a commercial 65 nm CMOS process to validate the performance. The chip die photo and area breakdown of the proposed SRAM MAC hardware accelerator is shown in Fig. 9 (a) and (c). The active area is 0.152 mm^2 , among which the ADC per each column occupies 80% of the total area. Note that the relatively large area overhead due to the ADC in the proposed architecture is mainly due to the small array size. Excluding the ADC block, the CCM circuit and the input current mirror occupy most of the area.

The table in Fig. 9(b) summarizes the major features of the proposed PIM architecture. The size of a single bit cell is $2.465 \times 1.465 \text{ } \mu\text{m}^2$. Under 1V supply voltage, the chip has been tested with 50 MHz clock signal. The output from the MAC operation along a single column was converted into the voltage level, which ranges from 50mV to 530mV based on the post-layout simulation. Finally, the input, weight, and output bit precision supported in the proposed design is 4-bit each.

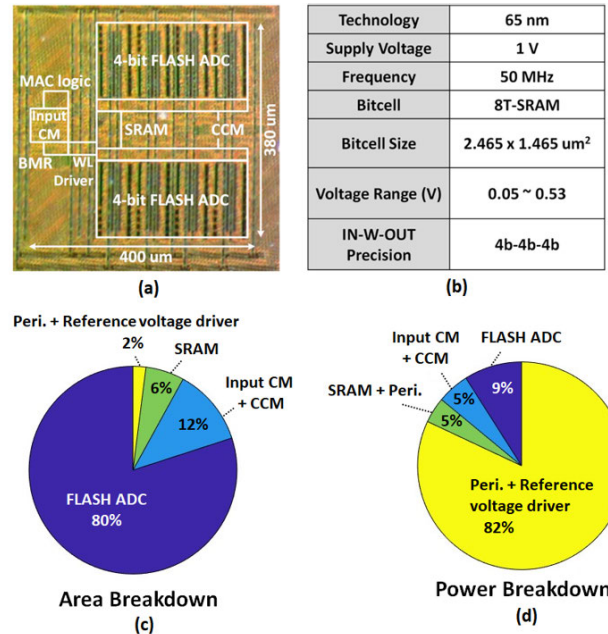


FIGURE 9. (a) Chip die photo, (b) performance summary of fabricated chip, (c) area breakdown and (d) power breakdown.

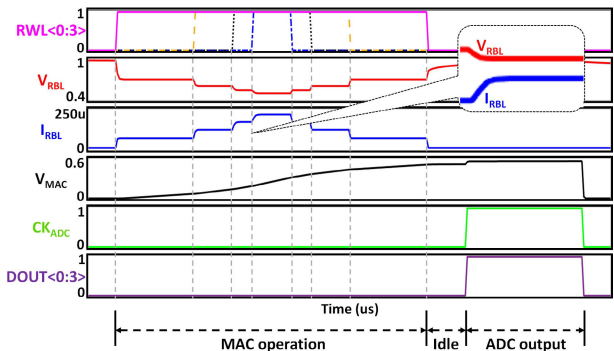


FIGURE 10. Timing diagram of MAC operation.

Fig. 9 (d) shows the power breakdown of the fabricated chip, and shows that reference voltage driver for FLASH ADC has been made over-spec and leaves room for energy improvement. After post layout simulation, we found out that the 97.9 percent of power consumption of reference voltage driver itself is redundant, which accounts for 80.3 percent of total power consumption of entire chip. So we lowered the power consumption of reference voltage driver this much, and it led to power saving of entire chip up to X4.83.

Fig. 10 shows the entire process of the proposed MAC operation. When $RWL<3:0>$ are turned on sequentially as illustrated in the previous section, the V_{RBL} and the corresponding I_{RBL} level change accordingly. The I_{RBL} charges the RBL capacitor to increase the V_{MAC} level, which is then converted to a 4-bit digital code by the ADC.

The results of the 4-bit precision MAC operation obtained from the post-layout simulation and the actual measurement

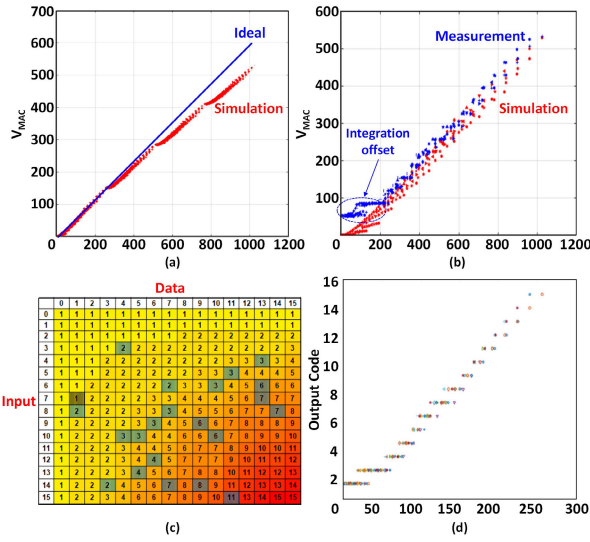


FIGURE 11. (a) simulated V_{MAC} linearity curve, (b) measured MAC linearity curve, (c) ADC output code occurrence table and (d) ADC output code based on the MAC operation output.

are shown in Fig. 11. First, the MAC calculation results along a single column in terms of output voltage (V_{MAC}) are shown in Fig. 11(a). Compared with the ideal graph, the simulation result shows a deviation as the multiplication output is expected to have a higher value due to the inherent nonlinearity of the transistor I-V characteristics.

Fig. 11(b) shows the MAC output voltage difference between the post-layout simulation result and the prototype chip measurement result. Note that there was an anomaly in our measurements that caused the V_{MAC} voltage to rise slightly when the output voltage was small, which we believe is caused by the ADC integration offset. Nevertheless, the measurement results show a reasonably good correlation with the simulation result.

The result of the MAC operation on the ADC output code is shown in Fig. 11(c) and (d), where the former table shows the most frequently observed ADC code depending on the different input and weight combinations, and the latter graph shows a graphical representation of the table. For the ADC output code table, the outputs are selected over multiple iterations ($\sim 1,000X$), and then colored from yellow to red as the output code increases. The green colored boxes in the table represent the low accuracy output code (less frequent than usual). However, the low accuracy code typically occurs at the boundary of the code change, which is quite understandable.

B. MACHINE LEARNING ALGORITHM ANALYSIS

Since input / weight / output bit precisions have been constrained to 4 bit to maximize energy benefits during machine learning inference task, it is crucial to analyze its accuracy performance. For this, we performed machine learning classification task for MNIST and CIFAR-10 inference task.

TABLE 1. Machine learning classification accuracy.

DATASET	MNIST			CIFAR-10		
MODEL	784-200-10 MLP			RESNET-18		
TYPE	S/W	QAT 4 b + Linear CCM curve	QAT 4 b + Nonlinear curve w/o CCM	S/W	QAT 4 b + Linear CCM curve	QAT 4 b + Nonlinear curve w/o CCM
ACCURACY	98.03%	98.02%	97.01% (-1.01%)	95.13%	94.93%	91.45% (-3.48%)

Common approach to applying machine learning models into MAC hardware accelerator is to quantize the model parameters to MAC hardware bit precision. Then it causes severe accuracy drop due to quantization errors. To resolve this issue, previous work [24] developed a technique know as Quantization Aware Training (QAT), which applies quantization itself into training procedure and prevents accuracy drop due to quantization.

To analyze the accuracy performance of our design, we utilized brevitas QAT framework [24], and quantized input, weight and output bit precision into 4 bit and performed training and inference task for MNIST and CIFAR-10 datasets. 784-200-10 MLP model used for MNIST dataset classification, and RESNET-18 model was used for CIFAR-10 dataset classification. As can be seen on Table 1, the accuracy drops of QAT 4 bit models fall within 0.2 percent for both classification tasks.

Also to analyze the linearity improvement of designed cascade current mirror, we added nonlinear phenomenon of bitcell current expressed in equation [2] into each MLP model and RESNET-18 model, and performed inference task. As can be seen on Table 1, the accuracy has been dropped by 1.01 percent for MNIST dataset classification and 3.48 percent for CIFAR-10 dataset classification. This result not only proves the efficacy of our cascade current mirror design, but also emphasizes the linearity of PIM architecture MAC operation with more complex DNN models.

To sum up, it can be concluded that constraints posed on bit precision enables energy efficiency improvements, with accuracy drop prevention with the aid of QAT and cascade current mirror.

C. PERFORMANCE COMPARISON WITH STATE-OF-THE-ART RESEARCHES

Finally, Table 2 summarizes and compares the performance of the proposed SRAM MAC hardware accelerator with the previous works with similar conditions. This work shows 0.29 GOPs of throughput and 1.05 TOPs/W of energy efficiency, which exhibits comparable performance metrics compared to the similar works. When the power consumption of reference voltage driver is reduced by upto 2.1 percent from the original one, energy efficiency can be improved upto 5.07 TOPs/W. Furthermore, this work proves that the proposed current mirror-based multi-bit precision MAC operator shows reasonably good linearity to be useful for the error-resilient neural network hardware accelerator.

TABLE 2. Performance comparison.

	[7]	[22]	[23]	[12]	This work
Process	65nm	180 nm	55nm	40 nm	65 nm
Supply Voltage	1.0 V	1.8 V	0.9 V	0.8 V	1 V
Memory Cell type	SRAM	SRAM	SRAM	SRAM	8T SRAM
Operation mode	Analog	Analog	Mixed	Digital	Analog
Operating frequency	100 - 400 MHz	20 MHz	285 MHz	200 MHz	50 MHz
IN/W/OUT bit precision	4 / 4 / 1	4 / 4 / 8	2 / 2 / 7	1-16 / 1-16 / 1-16	4 / 4 / 4
Array Size	512 x 256	128 x 16	64 x 64	40K x 40K	16 x 8
Throughputs (GOPs)	430	1.28	82.29	74	0.29
Energy efficiency (TOPs/W)	3.13	0.64	10.1	0.27 - 2.75	1.05 *5.07
Accuracy (MNIST)	-	97.5%	98.77	-	98.02 %

* When the reference voltage driver is optimized for power

Also, due to the controllability of the current mirror-based approach, increasing the bit resolution seems feasible without increasing the bit cell area. We also expect that higher performance can be achieved by optimizing the operation timing and optimizing the area through ADC share, which were conservatively approached in the prototype chip.

IV. CONCLUSION

In this work, we proposed 8T SRAM based process in memory (PIM) system with current mirror to offer a breakthrough in energy bottleneck incurred when DNN task is processed by conventional computing platform. We demonstrated 4 bit input, 4 bit weight and 4 bit output bit precision for multiple bit width configuration. Furthermore, MAC operation linearity has been improved for the additionally adopted cascaded current mirror stages and center-allocation of RWL timing pulses. Prototype chip was fabricated with TSMC 65nm GP process technology. The measurement result has proved the linearity of MAC operation. Throughput and energy efficiency of 0.29 GOPs and 1.05 TOPs/W have proved the system's efficacy when it is applied to common DNN tasks. Also, our design's efficacy in machine learning classification task has been proved with 98.02 percent accuracy for MNIST dataset classification task and 94.93 percent accuracy for CIFAR-10 dataset classification task.

REFERENCES

- [1] M. Horowitz, "computing's energy problem (and what we can do about it)," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2014, pp. 10–14.
- [2] A. Jaiswal, I. Chakraborty, A. Agrawal, and K. Roy, "8T SRAM cell as a multibit dot-product engine for beyond von Neumann computing," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 27, no. 11, pp. 2556–2567, Nov. 2019.
- [3] A. Agrawal, A. Jaiswal, C. Lee, and K. Roy, "X-SRAM: Enabling in-memory Boolean computations in CMOS static random access memories," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 65, no. 12, pp. 4219–4232, Dec. 2018.
- [4] Z.-R. Wang, Y. Li, Y.-T. Su, Y.-X. Zhou, L. Cheng, T.-C. Chang, K.-H. Xue, S. M. Sze, and X.-S. Miao, "Efficient implementation of Boolean and full-adder functions with 1T1R RRAMs for beyond von Neumann in-memory computing," *IEEE Trans. Electron Devices*, vol. 65, no. 10, pp. 4659–4666, Oct. 2018.
- [5] S. K. Gonugondla, M. Kang, and N. Shanbhag, "A 42pJ/decision 3.12TOPS/W robust in-memory machine learning classifier with on-chip training," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2018, pp. 490–492.
- [6] H. Kim, T. Yoo, T. T. Kim, and B. Kim, "Colonnade: A reconfigurable SRAM-based digital bit-serial compute-in-memory macro for processing neural networks," *IEEE J. Solid-State Circuits*, vol. 56, no. 7, pp. 2221–2233, Jul. 2021.
- [7] K. Ando, K. Ueyoshi, K. Orimo, H. Yonekawa, S. Sato, H. Nakahara, S. Takamaeda-Yamazaki, M. Ikebe, T. Asai, T. Kuroda, and M. Moto-mura, "BREIN memory: A single-chip binary/ternary reconfigurable in-memory deep neural network accelerator achieving 1.4 Tops at 0.6 W," *IEEE J. Solid-State Circuits*, vol. 53, no. 4, pp. 983–994, Apr. 2018.
- [8] J. Wang, X. Wang, C. Eckert, A. Subramaniam, R. Das, D. Blaauw, and D. Sylvester, "A 28-nm compute SRAM with bit-serial logic/arithmetic operations for programmable in-memory vector computing," *IEEE J. Solid-State Circuits*, vol. 55, no. 1, pp. 76–86, Jan. 2020.
- [9] C. Yu, T. Yoo, K. T. C. Chai, T. T. Kim, and B. Kim, "A 65-nm 8T SRAM compute-in-memory macro with column ADCs for processing neural networks," *IEEE J. Solid-State Circuits*, vol. 57, no. 11, pp. 3466–3476, Nov. 2022.
- [10] V. Sharma, J.-E. Kim, H. Kim, L. Lu, and T. T. Kim, "A reconfigurable 16Kb AND8T SRAM macro with improved linearity for multibit compute-in-memory of artificial intelligence edge devices," *IEEE J. Emerg. Sel. Topics Circuits Syst.*, vol. 12, no. 2, pp. 522–535, Jun. 2022.
- [11] L. Cavigelli and L. Benini, "Origami: A 803-GOPs/W convolutional network accelerator," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 27, no. 11, pp. 2461–2475, Nov. 2017.
- [12] B. Moons and M. Verhelst, "An energy-efficient precision-scalable ConvNet processor in 40-nm CMOS," *IEEE J. Solid-State Circuits*, vol. 52, no. 4, pp. 903–914, Apr. 2017.
- [13] Y.-H. Chen, T. Krishna, J. S. Emer, and V. Sze, "Eyeriss: An energy-efficient reconfigurable accelerator for deep convolutional neural networks," *IEEE J. Solid-State Circuits*, vol. 52, no. 1, pp. 127–138, Jan. 2017.
- [14] X. Si, J.-J. Chen, Y.-N. Tu, W.-H. Huang, J.-H. Wang, Y.-C. Chiu, W.-C. Wei, S.-Y. Wu, X. Sun, R. Liu, S. Yu, R.-S. Liu, C.-C. Hsieh, K.-T. Tang, Q. Li, and M.-F. Chang, "A twin-8T SRAM computation-in-memory unit-macro for multibit CNN-based AI edge processors," *IEEE J. Solid-State Circuits*, vol. 55, no. 1, pp. 189–202, Jan. 2020.
- [15] Z. Jiang, S. Yin, J.-S. Seo, and M. Seok, "C3SRAM: An in-memory-computing SRAM macro based on robust capacitive coupling computing mechanism," *IEEE J. Solid-State Circuits*, vol. 55, no. 7, pp. 1888–1897, Jul. 2020.
- [16] Z. Chen, Z. Yu, Q. Jin, Y. He, J. Wang, S. Lin, D. Li, Y. Wang, and K. Yang, "CAP-RAM: A charge-domain in-memory computing 6T-SRAM for accurate and precision-programmable CNN inference," *IEEE J. Solid-State Circuits*, vol. 56, no. 6, pp. 1924–1935, Jun. 2021.
- [17] A. Biswas and A. P. Chandrakasan, "CONV-SRAM: An energy-efficient SRAM with in-memory dot-product computation for low-power convolutional neural networks," *IEEE J. Solid-State Circuits*, vol. 54, no. 1, pp. 217–230, Jan. 2019.
- [18] M. Ali, A. Jaiswal, S. Kodge, A. Agrawal, I. Chakraborty, and K. Roy, "IMAC: In-memory multi-bit multiplication and accumulation in 6T SRAM array," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 67, no. 8, pp. 2521–2531, Aug. 2020.
- [19] J. Mu, H. Kim, and B. Kim, "SRAM-based in-memory computing macro featuring voltage-mode accumulator and row-by-row ADC for processing neural networks," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 69, no. 6, pp. 2412–2422, Jun. 2022.
- [20] D. Kushwaha, A. Sharma, N. Gupta, R. Raj, A. Joshi, J. Mishra, R. Kohli, S. Miryala, R. Joshi, S. Dasgupta, and A. Bulusu, "A 65 nm compute-in-memory 7T sram macro supporting 4-bit multiply and accumulate operation by employing charge sharing," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2022, pp. 1556–1560.
- [21] Z. Lin, H. Zhan, Z. Chen, C. Peng, X. Wu, W. Lu, Q. Zhao, X. Li, and J. Chen, "Cascade current mirror to improve linearity and consistency in SRAM in-memory computing," *IEEE J. Solid-State Circuits*, vol. 56, no. 8, pp. 2550–2562, Aug. 2021.

- [22] V. T. Nguyen, J.-S. Kim, and J.-W. Lee, "10T SRAM computing-in-memory macros for binary and multibit MAC operation of DNN edge processors," *IEEE Access*, vol. 9, pp. 71262–71276, 2021.
- [23] Y.-C. Chiu, Z. Zhang, J.-J. Chen, X. Si, R. Liu, Y.-N. Tu, J.-W. Su, W.-H. Huang, J.-H. Wang, W.-C. Wei, J.-M. Hung, S.-S. Sheu, S.-H. Li, C.-I. Wu, R.-S. Liu, C.-C. Hsieh, K.-T. Tang, and M.-F. Chang, "A 4-Kb 1-to-8-bit configurable 6T SRAM-based computation-in-memory unit-macro for CNN-based AI edge processors," *IEEE J. Solid-State Circuits*, vol. 55, no. 10, pp. 2790–2801, Oct. 2020.
- [24] A. Pappalardo, "Xilinx/brevitas," *Tech. Rep.*, 2023, doi: [10.5281/zenodo.3333552](https://doi.org/10.5281/zenodo.3333552).



DONG-HYUN SHIN (Graduate Student Member, IEEE) received the B.S. degree from the School of Electrical and Electronics Engineering, Chung-Ang University (CAU), Seoul, South Korea, in 2022. He is currently pursuing the M.S. degree in intelligent semiconductor engineering. His research interest includes power management circuits (PMICs).



KEUN-YONG CHUNG received the B.S. and M.S. degrees from the School of Electrical and Electronics Engineering, Chung-Ang University (CAU), Seoul, South Korea, in 2021 and 2023, respectively. He is currently with Samsung Electronics, South Korea.



KWANG-HYUN BAEK (Senior Member, IEEE) received the B.S. and M.S. degrees from Korea University, Seoul, South Korea, in 1990 and 1998, respectively, and the Ph.D. degree in electrical engineering from the University of Illinois at Urbana–Champaign (UIUC), IL, USA, in 2002. From 2000 to 2006, he was a Senior Scientist with the Department of High-Speed Mixed-Signal ICs, Rockwell Scientific Company, formerly Rockwell Science Center (RSC), Thousand Oaks, CA, USA.

At RSC, he was involved in the development of high-speed data converters (ADC/DAC) and direct digital frequency synthesizers (DDFS). He was with Samsung Electronics, from 1990 to 1996. Since 2006, he has been with the School of Electrical and Electronics Engineering, Chung-Ang University (CAU), Seoul, where he is a Faculty Member. His research interests include high-performance analog and digital circuits, such as low-power ADCs, high-speed DACs, hybrid frequency synthesizers (PLLs, DDFSs), high-speed interface circuits (CDRs and SerDes), PMIC, and near threshold-voltage (NTV) circuits.



HONGGU KIM (Graduate Student Member, IEEE) received the B.S. degree from the School of Electrical and Electronics Engineering, Chung-Ang University (CAU), Seoul, South Korea, in 2022. He is currently pursuing the M.S. degree in intelligent semiconductor engineering. His research interests include neuromorphic hardware and software co-optimization.



YERIM AN (Graduate Student Member, IEEE) received the B.S. degree from the School of Electrical and Electronics Engineering, Tech University of Korea (TUKOREA), South Korea, in 2022. She is currently pursuing the M.S. degree in intelligent semiconductor engineering with Chung-Ang University (CAU). Her research interest includes design process-in-memory (PIM) architecture.



KIHO SEONG (Graduate Student Member, IEEE) received the B.S. and M.S. degrees from the School of Electrical and Electronics Engineering, Chung-Ang University (CAU), Seoul, South Korea, in 2018 and 2020, respectively. He is currently pursuing the Ph.D. degree in electrical and electronics engineering. His research interest includes high-speed and high-resolution analog-to-digital converter (ADC).



YONG SHIM (Member, IEEE) received the B.S. and M.S. degrees in electronics engineering from Korea University, in 2004 and 2006, respectively, and the Ph.D. degree from the School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN, USA, in 2018. He was a Memory Interface Designer with Samsung Electronics, Hwaseong, from 2006 to 2013. At Samsung, he has worked on the design and development of a memory interface for synchronous DRAMs (DDR1 and DDR4). He is currently an Assistant Professor with Chung-Ang University. Prior to joining Chung-Ang University, in 2020, he was an SRAM Designer with Intel Corporation, Hillsboro, OR, USA, from 2018 to 2020, where he was involved in designing circuits for super-scaled next-generation SRAM cache design. His research interests include neuromorphic hardware and algorithms, in-memory computing, robust memory interface design, and emerging devices (RRAM, MRAM, and STO) based unconventional computing models.

...